PRÁCTICA 2 DIVISIÓN

TÍTULO

Division

NOMBRE Mezo Sandoval Claudia FECHA

25/02/2025

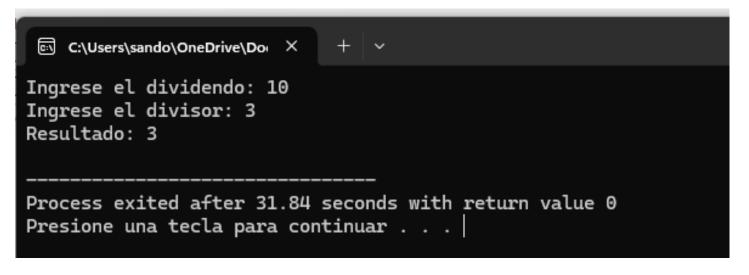
Dados dos números enteros dividendo y divisor, divida dos números enteros sin utilizar la multiplicación, la división y el operador módulo.

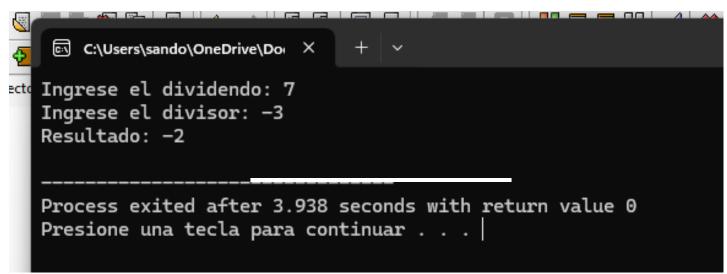
Código

```
#include <iostream>
     #include <climits> // Para INT MAX e INT MIN
     #include <cmath> // Para abs()
 4
 5
     using namespace std;
 7 int divide(int dividend, int divisor) {
         // Manejo del caso especial de desbordamiento
 9 🗀
         if (dividend == INT MIN && divisor == -1) {
             return INT MAX; // Evita overflow (2^31 no cabe en int)
10
11
12
13
         // Determinar si el resultado es negativo
14
         bool negative = (dividend < 0) ^ (divisor < 0); // XOR Lógico
15
         // Convertir a valores positivos usando long long para evitar desbordamiento
16
17
         long long absDividend = abs((long long)dividend);
         long long absDivisor = abs((long long)divisor);
18
19
         int quotient = 0;
20
21
         // Restas exponenciales usando desplazamiento de bits
22
23 -
         while (absDividend >= absDivisor) {
             long long temp = absDivisor, multiple = 1;
24
25
26 -
             while (absDividend >= (temp << 1)) {
                 temp <<= 1; // Multiplicamos por 2 (bit shift a la izquierda)
27
28
                 multiple <<= 1;
29
30
             absDividend -= temp;
31
             quotient += multiple;
32
33
34
```

```
34
         // Devolver el resultado con el signo correcto
35
         return negative ? -quotient : quotient;
36
37
38
     // Función principal con entrada del usuario
39
40 -
     int main() {
         int dividend, divisor;
41
42
         // Pedir al usuario que ingrese los valores
43
         cout << "Ingrese el dividendo: ";
44
         cin >> dividend;
45
46
         cout << "Ingrese el divisor: ";
47
48
         cin >> divisor;
49
50
         // Manejo de error si el divisor es 0
         if (divisor == 0) {
51 -
              cout << "Error: No se puede dividir por cero." << endl;
52
              return 1; // Código de error
53
54
55
56
         // Se muestra el resultado
         cout << "Resultado: " << divide(dividend, divisor) << endl;</pre>
57
58
59
         return 0;
60
```

Resultado





https://github.com/claudia2025/practical_numero_palindromo